

WHAT IS CLAIMED IS:

1 1. A lightweight alarm manager running in a Web browser, to be applied to a computer
2 connected to NMS (Network Management System) over network, comprising:
3 a header frame fixing a title label of the alarm manager;
4 a data frame receiving alarm information from the NMS through the network and managing
5 the alarm information in XML (Extensible Markup Language) format; and
6 a contents frame having dynamic HTML (Hypertext Markup Language) for reading the alarm
7 information being managed in the data frame and providing a user with the alarm information in a
8 data table system.

1 2. The alarm manager of claim 1, wherein the contents frame handles a table object of a
2 HTML provided by the Web browser, and provides a GUI (Graphical User Interface).

1 3. The alarm manager of claim 1, wherein the contents frame provides the alarm information
2 that is comprised of {severity} of the alarm, {eventtime} when the alarm is raised, alarm ID
3 {alarm_id}, components of network equipment, {dn}, where the alarm is raised, and contents of the
4 alarm {contents}.

1 4. A service method of a lightweight alarm manager running in a Web browser, to be applied
2 to a computer connected NMS (Network Management System) through network, the service method

3 comprising the steps of:

4 receiving a request from a user to use the alarm manager;

5 creating a header frame, a contents frame, and a data frame on the Web browser in the alarm
6 manager, in response to an alarm manager service request from a user;

7 requesting, at the data frame, that the NMS provide alarm information periodically to the data
8 frame;

9 managing the alarm information in the alarm manager when the alarm information is received
10 by the data frame;

11 periodically checking, by the contents frame made up of dynamic HTML, whether the alarm
12 information in the data frame is being properly managed;

13 accessing and obtaining, by the contents frame, the alarm information being managed by the
14 data frame;

15 constructing a data table of the alarm information being managed by the data frame; and

16 displaying the alarm information to the user.

1 5. The method of claim 4, wherein the requesting step is comprised of the sub-steps of:

2 requesting the NMS connected to the data frame through the network to provide alarm
3 information periodically;

4 receiving alarm information in a XML format from the NMS; and

5 managing the received alarm information in the XML format.

1 6. The method of claim 4, wherein the accessing and obtaining, constructing and displaying
2 steps comprise:

3 obtaining, by the contents frame, the alarm information in XML format from the NMS;
4 managing, by the data frame, the received alarm information in the data frame;
5 simply adding a row to a table object, by the contents frame, using attributes of the table
6 object of a HTML provided by the Web browser; and
7 displaying the alarm information being obtained using the table object.

1 7. The method of claim 5, wherein the accessing and obtaining, constructing and displaying
2 steps comprise:

3 obtaining, by the contents frame, the alarm information in XML format from the NMS;
4 managing, by the data frame, the received alarm information in the data frame;
5 simply adding a row to a table object, by the contents frame, using attributes of the table
6 object of a HTML provided by the Web browser; and
7 displaying the alarm information being obtained using the table object.

1 8. The method of claim 6, wherein the step for adding a row by the contents frame comprises
2 sub-steps of:

3 checking when a number of current rows in the table object provided by the Web browser
4 is greater than a predetermined number of rows;
5 deleting the oldest record when the number of current rows in the table object provided by

the Web browser is greater than the predetermined number;

creating a new row in the table object comprising the alarm information read by the contents frame; and

displaying the alarm information of the table object when the number of current rows in the table object provided by the Web browser is not greater than the predetermined number of rows to be maintained.

9. The method of claim 7, wherein the step for adding a row by the contents frame comprises sub-steps of:

checking when a number of current rows in the table object provided by the Web browser is greater than a predetermined number of rows;

deleting the oldest record when the number of current rows in the table object provided by the Web browser is greater than the predetermined number;

creating a new row in the table object comprising the alarm information; and

displaying the alarm information of the table object when the number of current rows in the table object provided by the Web browser is not greater than the predetermined number of rows to be maintained.

10. A method of providing alarm information to a lightweight alarm manager running in a Web browser, the method comprising the steps of:

receiving, at NMS (Network Management System), an alarm information request from an

alarm manager via a network;

confirming, at the NMS, session information related to the alarm manager, and obtaining time information used to form an alarm information packet to be transmitted to the alarm manager;

obtaining, at the NMS, additional alarm information from a database in the NMS, the additional information being based on the time information, the additional information being added to the packet;

converting the packet into XML format at the NMS, and

transmitting the packet of alarm information in XML format to the alarm manager.

11. The method of claim 10, further comprising the step of managing, at the NMS, the session information related to each of the alarm managers via a checkSession thread.

12. A method of providing alarm information to a lightweight alarm manager running in a Web browser, the method comprising the steps of:

receiving, at NMS (Network Management System), an alarm information request from an alarm manager via a network;

creating, at the NMS, a service thread for providing alarm information at a request of the alarm manager, and a checkSession thread for managing session information related to the alarm manager;

determining, at the service thread, whether session information related to the alarm manager exists is present in the NMS, and when the session information is not present, creating a new session

10 information, and when the session information is present, extracting a final search alarm occurrence
11 time from the session information;

12 based on the alarm occurrence time, obtaining, via the service thread, additional alarm
13 information by searching a database in the NMS and updating the session information accordingly
14 with information found in the database, said additional alarm information being based on the alarm
15 occurrence time;

16 converting, at the service thread, the alarm information into an XML format, and transmitting
17 the alarm information to the alarm manager as a response to the request; and

18 checking, at the checkSession thread, an update date of the session info, and deleting the
19 session information when the session information is not valid.

1 13. The method of claim 12, wherein the session information has information about the
2 alarm manager that sent the request, information about a user using the alarm manager, and
3 information about time of final alarm occurrence of the alarm manager.

1 14. A NMS(Network Management System) server adapted to serve a plurality of alarm
2 managers, the NMS server comprising:

3 a JSP(Java Server Page) engine comprising a makeXML JSP adapted to transmit XML data
4 to a data frame of one of said plurality of lightweight alarm managers, and a JSP context adapted to
5 store session information about said plurality of lightweight alarm managers; and

6 a database comprising alarm information.

1 15. The NMS server of claim 14, wherein the makeXML JSP provides a service thread that
2 is adapted to offer alarm information at the request of each of the plurality of lightweight alarm
3 managers, and a checkSession thread adapted to manage (or check) an existence of each of the
4 plurality of lightweight alarm managers.

1 16. The NMS server of claim 15, wherein the makeXML JSP regularly receives a HTTP
2 request from each of the lightweight managers, and confirms a final date and time to extract new
3 alarm information from the session information in the JSP Context.

1 17. The NMS server of claim 14, wherein the makeXML JSP , using a final date and time
2 as a starting point, queries data from the database and constructs a XML document representing the
3 data from the database and transmits the XML document to a data frame of one of the plurality of
4 lightweight alarm managers.

1 18. The NMS server of claim 14, wherein the JSP context comprising the stored session
2 information is comprised of NMS user information using the plurality of alarm managers, and
3 information about a time when a corresponding one of said plurality of alarm managers raises a final
4 alarm.